IN THE CLAIMS:

1. (currently amended) An application management system comprising:

a first computer configured to host an application;

a user terminal;

a communication network, said first computer ~~communicatively associated with~~ coupled to said user terminal through said communication network;

an computer-executable first application, wherein at least a portion of said first application is executed by said first computer;

a computer-executable second application; and

a computer-executable keep-alive function, to collect first application timeout information related to said first application to formulate a keep-alive input based on the collected first application timeout information; and to transmit said keep-alive input to said first computer.

~~a computer-executable timeout function hosted by said first computer, said timeout function configured to cause said first computer to terminate said application if a keep-alive input is not received by said first computer within a timeout period; and~~

~~a computer-executable keep-alive function hosted by said user terminal, said keep-alive function configured to cause said user terminal to transmit a message to said first computer within a period based on said timeout period.~~

Claim 2. (cancelled).

3. (currently amended) The application management system according to claim [[2]] 1, wherein said first computer is a server, and said user terminal establishes communication with said ~~first computer~~ server by sending said ~~first computer~~ server a

session initiation request.

Claim 4. (cancelled).

5. (currently amended) The application management system according to claim [[4]] 1, wherein

said keep-alive input has a content and a format, and

at least one of said content and said format is based on said timeout information collected by said keep-alive function.

6. (currently amended) The application management system according to claim [[4]] 1, wherein said first application timeout information includes information related to [[the]] a timeout period.

7. (currently amended) The application management system according to claim [[4]] 1, wherein:

said first computer maintains an application timeout clock, [[said]] a timeout function terminating said application when said timeout clock reaches a specified state and said application timeout clock being reset when said first computer receives [[a]] the keep-alive input, and

said timeout information includes information related to a current state of said timeout clock.

8. (currently amended) The application management system according to claim [[2]] 1, wherein said keep-alive function maintains a keep-alive function timeout clock and said keep-alive input is transmitted to said first computer when said keep-alive function timeout clock reaches a specified state.

9. (currently amended) The application management system according to claim

8, wherein:

said first computer maintains an application timeout clock, [[said]] a timeout function terminating said first application when said timeout clock reaches a specified state and said application timeout clock being reset when said first computer receives [[a]] the keep-alive input,

said timeout information ~~including~~ includes information related to a current state of said application timeout clock, and

said keep-alive function timeout clock is set based on said timeout information.

10. (original) An application management system according to claim 9, wherein said keep-alive function timeout clock is reset when the application timeout clock has been reset.

11. (currently amended) An application management system according to claim 9, wherein said keep-alive function timeout clock is reset when the user performs an action to keep said application alive.

12. (currently amended) The application management system according to claim [[2]] 1, wherein said keep-alive function causes a query to be sent to a user seeking authorization from said user to transmit said keep-alive input to said first computer.

Claims 13 - 29 (cancelled).

30. (currently amended) A method of managing ~~an application~~ a plurality of applications, ~~at least a portion of said application being executed by a first computer and said application being subject to termination by said first computer if said first computer does not receive a keep-alive input within an application~~

~~timeout period~~, said method comprising:

executing a first application, at least a portion of the first application being executed by a first computer and at least a second portion of the first application being executed at a user terminal, the first computer and the user terminal being coupled via a network;

executing a second application at the user terminal;

~~by a keep-alive function hosted by a user terminal,~~ collecting, by a keep-alive function, first application timeout information related to said first application;

formulating a keep-alive input based on the collected timeout information; and

transmitting said keep-alive input from said user terminal to said first computer.

31. (currently amended) The method of ~~managing an application according to~~ claim 30, wherein said application timeout information includes information related to an application timeout clock maintained by said first computer and related to said first application.

32. (currently amended) The method of ~~managing an application according to~~ claim 31, wherein said keep-alive input is transmitted based on a state of said application timeout clock.

33. (currently amended) The method ~~of managing an application according to~~ claim 31, further including, by said keep-alive function, maintaining a keep-alive function timeout clock related to said first application timeout clock, and wherein said keep-alive input is transmitted based on a state of said keep-alive function timeout clock.

34. (cancelled).

35. (currently amended) The method of claim 30, wherein said application timeout includes information related to ~~one of~~ a required content or a required format for said keep-alive input, and said keep alive input is formulated based on said application timeout information.

Claim 36 (cancelled).

37. (new) The method of claim 30, wherein said application timeout information includes information related to a required format for said keep-alive input, and said keep-alive input is formulated based on said application timeout information.

38. (new) The method of claim 30, further including transmitting a keep-alive information request to the first computer in order to collect the first application timeout information.

39. (new) The method of claim 38, wherein the keep-alive information request is a request identifying environmental variables or parameters associated with the time-out of the first application and a response includes the environmental variables or parameters.

40. (new) The method of claim 39, wherein the keep-alive function sorts through the environmental variables or parameters to identify timeout parameters.

41. (new) The method of claim 39, further including sorting, by the keep-alive function, through the environmental variables or parameters to identify likely timeout parameters and determining whether multiple likely timeout parameters are present.

42. (new) The method of claim 41, further including selecting one of the multiple likely timeout parameters having a smallest value.

43. (new) The application management system of claim 1, wherein the

keep-alive function is housed on a computing device that is not the first computer or the user terminal.

44. (new) The application management system of claim 1, wherein the keep-alive function is housed on the user terminal.

45. (new) The application management system of claim 1, wherein at least a portion of the second application is executed by said first computer.

46. (new) The application management system of claim 1, wherein the second application is executed at the user terminal.

47. (new) The application management system of claim 1, wherein the keep-alive function transmits a keep-alive information request to the first computer in order to collect the first application timeout information.

48. (new) The application management system of claim 1, wherein the first application is a downloading web page application and the keep-alive input transmitted to the first computer includes instructions to refresh the content of a web page.

49. (new) The application management system of claim 1, wherein the keep-alive function transmits a keep-alive information request to the user terminal in order to collect the first application timeout information.